

REMARKS

Claims 1-35 are pending in the application, of which claims 1-35 stand rejected.

STATEMENT UNDER 1.133(b)

The undersigned representative would like to thank the Examiner for the courtesy of the telephone interview of November 18, 2005 in which the undersigned and Examiner Rampuria participated. During the interview it was agreed that the “final-action” paragraph at page 3, paragraph 6 was not relevant to the instant non-final office action and should be disregarded.

REJECTIONS UNDER 35 U.S.C. 103

Claims 1 and 2 stand rejected under 35 U.S.C. 103 as being unpatentable “over US Patent No. 6,144,965 to Oliver (hereinafter called Oliver) in view of US Patent No. 6,223,342 to George (hereinafter, George).” The Office Action states in relevant part that “Oliver does not explicitly disclose providing a function for automatically converting a smart pointer to an object of a first class to a smart pointer to an object of a second class. However, George discloses in an analogous computer system providing a function for automatically converting a smart pointer to an object of a first class to a smart pointer to an object of a second class (col. 6, lines 40-60... The conversion function is inherent to George system since the system has three smart pointers.”

Applicant agrees with the Examiner that Oliver fails to disclose at least the claimed feature of “providing a function for automatically converting a smart pointer to an object of a first class to a smart pointer to an object of a second class.” However, Applicant respectfully disagrees with the outstanding rejection of claims 1 and 2 for at least the reason that George fails to disclose the claimed feature acknowledged to be absent from Oliver.

Specifically, Applicant respectfully points out that the text of George relied upon in the Office Action at column 6, lines 40-60 fails to disclose or suggest the existence of a smart pointer pointing to an object of a first class and a smart pointer pointing to an object of a second class. Rather, George merely discloses the existence of “three smart pointers 14, 16 and 18” that “contain the identifications of the memory space for the three attributes of the object.[”] (Emphasis Added. Column 6, lines 41-42.) Thus, the three smart pointers 14, 16 and 18 of George point to a single object. Pointers that point to a single object are not, and cannot disclose

or suggest, pointers that point to an object of a first class and a second class. There is only one object in George to which the three smart pointers 14, 16 and 18 point. A single object is not inherently an object of a first class an object of a second class. Thus, the cited text of George fails to disclose Applicant's claimed features of "an object of a first class" and "an object of a second class" as recited in claim 1. Moreover, George cannot therefore disclose Applicant's claimed features of "a smart pointer to an object of a first class" and "a smart pointer to an object of a second class" as recited in claim 1. Hence, the cited text in George fails to disclose, or even address the situation, in which one has smart pointers that point to "an object of a first class" and smart pointers that point to "an object of a second class", as recited in Applicant's claim 1. Consequently, there is no need or suggestion to provide function for automatically converting a smart pointer to an object of a first class to a smart pointer to an object of a second class," since all three smart pointers 14, 16 and 18 point to a single object. The mere existence of more than one smart pointer does not automatically lead to the conclusion that the cited text inherently discloses Applicant's claimed feature of "providing a function for automatically converting a smart pointer to an object of a first class to a smart pointer to an object of a second class." Thus, the assertion in the Office Action that the "conversion function is inherent to George system since the system has three smart pointers" is misplaced, and the cited text in George fails to disclose implicitly or inherently Applicant's claimed feature of a "providing a function for automatically converting a smart pointer to an object of a first class to a smart pointer to an object of a second class."

Accordingly, for at least these reasons, Applicant respectfully requests that the Examiner withdraw the rejection of claim 1, as well as claim 2 which depends therefrom.

REJECTIONS UNDER 35 U.S.C. 102

Claims 3-35 stand rejected under 35 U.S.C. 102(e) "as being anticipated by US Patent No. 6,144,965 to Oliver (hereinafter called Oliver)."

With regard to the rejection of claim 3, the Office Action states in relevant part that "Oliver disclose:... providing a comparison means for comparing the value of the next pointer to the value of the memory location of the smart pointer in which the selected next pointer is

included, whereby a determination can be made if the ring contains more than one smart pointer (col. 5, lines 34-37 ...”. (Office Action, pages 6-7). Applicant respectfully disagrees.

The text of Oliver cited in the Office Action does not disclose Applicant’s claimed feature of “providing a function for automatically converting a smart pointer to an object of a first class to a smart pointer to an object of a second class...providing a comparison means for comparing the value of the next pointer to the value of the memory location of the smart pointer in which the selected next pointer is included, whereby a determination can be made if the ring contains more than one smart pointer” as recited in claim 3. (Emphasis Added.) In stark contrast, the text relied upon in the Office Action at column 5, lines 34-37 of Oliver states that the comparison is made between the value of the next pointer and the previous pointer. The “value of the memory location of the smart pointer in which the selected next pointer is included” is not even considered or mentioned in the cited text of Oliver. It is not used in the Oliver test. Only the next and previous pointers are used. Specifically, in Oliver the previous pointer is used for comparison to the next pointer. Hence, Oliver clearly fails to disclose at least Applicant’s claimed feature of “comparing the value of the next pointer to the value of the memory location of the smart pointer in which the selected next pointer is included.” For at least this reason, the rejection of claim 3 in view of Oliver is deficient, as Oliver fails to disclose each and every element recited in claim 3. Accordingly, since Oliver fails to disclose each and every element of claim 3, Applicant respectfully requests that the Examiner withdraw the rejection of claim 3, as well as claims 4-12 which depend respectively therefrom.

Applicant further respectfully submits that the above arguments relating to the allowability of claim 3 over Oliver also provide a basis for allowing independent claims 13, 23, and 35 for the following reasons.

Regarding claim 13, claim 13 recites “providing a comparison means for comparing the value of the memory location of the smart pointer to the value of the next pointer of the smart pointer, to provide a determination whether the linked list contains only the smart pointer” (Emphasis Added.) That is, claim 13 recites comparing “the value of the next pointer of the smart pointer” to “the value of the memory location of the smart pointer”. In other words, the comparison recited in the quoted text of claim 13 determines whether the next pointer of a particular smart pointer points to the smart pointer itself, i.e. does the pointer point to itself,

which is a correct test for whether there is only one pointer on a ring. As stated above with respect to claim 3, the “value of the memory location of the smart pointer in which the selected next pointer is included” is not even considered or mentioned in the cited text of Oliver. Thus, Oliver fails to disclose at least Applicant’s claimed comparison means “for comparing the value of the memory location of the smart pointer to the value of the next pointer of the smart pointer”, as recited in claim 13. Accordingly, since Oliver fails to disclose each and every element of claim 13, Applicant respectfully requests that the Examiner withdraw the rejection of claim 13, as well as claims 14-22 which depend respectively therefrom.

Regarding claim 23, claim 23 recites “comparing the value of the memory location of a selected smart pointer giving up its association with the memory-resident element to the value of the next pointer of the selected smart pointer, to provide a determination whether the linked list contains only the selected smart pointer” (Emphasis Added.) That is, claim 23 recites a comparison using “the value of the memory location of a selected smart pointer”. In a similar fashion as stated above with respect to claim 3, the “value of the memory location of the smart pointer” is not even considered or mentioned in the cited text of Oliver. Thus, Oliver fails to disclose at least Applicant’s claimed comparison means for “comparing the value of the memory location of a selected smart pointer giving up its association with the memory-resident element to the value of the next pointer of the selected smart pointer”, as recited in claim 23. Accordingly, since Oliver fails to disclose each and every element of claim 23, Applicant respectfully requests that the Examiner withdraw the rejection of claim 23, as well as claims 25-34 which depend respectively therefrom.

Regarding claim 35, claim 35 recites “comparing the value of the previous pointer to the value of the memory location of the smart pointer in which the selected previous pointer is included, whereby a determination can be made if the ring contains more than one smart pointer” (Emphasis Added.) That is, claim 35 recites a comparison using “the value of the memory location of the smart pointer”. In a similar fashion as stated above with respect to claim 3, the “value of the memory location of the smart pointer” is not even considered or mentioned in the cited text of Oliver. Thus, Oliver fails to disclose at least Applicant’s claimed comparison means for “comparing the value of the memory location of a selected smart pointer giving up its association with the memory-resident element to the value of the next pointer of the selected

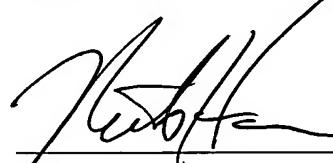
Application No.: 10/079,928
Docket No.: 3434-P02437US1

Art Unit: 2124
Examiner: Satish Rampuria

smart pointer", as recited in claim 35. Accordingly, since Oliver fails to disclose each and every element of claim 35, Applicant respectfully requests that the Examiner withdraw the rejection of claim 35.

In view of the foregoing amendments and remarks, it is believed that the claims in this application are now in condition for allowance. Early and favorable reconsideration is respectfully requested. The Examiner is invited to telephone the undersigned in the event that a telephone interview will advance prosecution of this application.

Respectfully submitted,



Niels Haun

PTO Reg. No. 48,488

DANN DORFMAN HERRELL & SKILLMAN
A Professional Corporation
1601 Market Street, Suite 2400
Philadelphia, PA 19103
Phone: (215) 563-4100
Fax: (215) 563-4044